5. Packets and frames

📡 Transport Layer & TCP/UDP

1. What is the Purpose of the Transport Layer in the OSI Model?

- **Layer 4 in OSI model** — bridges the gap between Network layer (IP) and Application layer protocols.
- Key functions:
  - **End-to-end communication:** Provides logical communication between apps on source and destination hosts.
  - **Segmentation and reassembly:** Breaks large data into segments for transmission and reassembles at receiver.
  - **Flow control:** Prevents sender from overwhelming receiver.
  - **Error detection and recovery:** Detects lost or corrupted data and triggers retransmissions.
  - **Multiplexing:** Supports multiple communication streams via ports.

2. Differences Between TCP and UDP

Feature	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
Type	Connection-oriented	Connectionless
Reliability	Reliable (ACKs, retransmissions, sequencing)	Unreliable (no ACKs, no retransmissions)
Flow Control	Yes (window-based)	No
Error Detection	Yes (checksum + retransmission)	Checksum only, no retransmission
Ordering	Guaranteed in-order delivery	No guarantee
Overhead	Higher due to connection management and error recovery	Lower overhead, faster
Use Cases	File transfers, web browsing, email	Streaming, VoIP, gaming

3. Differences Between Packets and Frames

Aspect	Packet	Frame
OSI Layer	Network Layer (Layer 3)	Data Link Layer (Layer 2)

Aspect	Packet	Frame
Contents	Contains IP header + payload (e.g., TCP/UDP segment)	Contains MAC header + payload (usually IP packet)
Scope	Logical unit of data sent across networks	Physical unit of data sent between nodes on same network segment
Role	Routes data between hosts	Moves data between devices on local LAN

---

4. How Does TCP Ensure Reliable End-to-End Communication Over an Unreliable Network?

- Uses **sequence numbers** to track data segments.
- Sender waits for **ACKnowledgements** from receiver to confirm receipt.
- Implements **retransmission** if ACK not received within timeout.
- Uses **checksums** for error detection.
- Performs **flow control** using sliding window protocol to avoid congestion.
- Uses **congestion control algorithms** (e.g., slow start, congestion avoidance) to optimize network usage.
- Ensures **in-order delivery** by buffering out-of-order packets.

---

5. What is the Purpose of the TCP Three-Way Handshake Process?

- Establishes a **reliable connection** before data transfer.
- Steps:
  1. **SYN:** Client sends a SYN packet to initiate connection.
  2. **SYN-ACK:** Server acknowledges with SYN-ACK to confirm receipt and willingness to connect.
  3. **ACK:** Client sends ACK to finalize connection.
- Synchronizes **sequence numbers** between sender and receiver for reliable communication.
- Prevents **half-open connections** and ensures both parties are ready.

---

6. Advantages and Disadvantages of TCP and UDP

Protocol	Advantages	Disadvantages
TCP	Reliable, ordered data transfer, error recovery	Higher latency and overhead due to connection setup and management
UDP	Faster, lower overhead, suitable for real-time apps	No guarantee of delivery, ordering, or error correction

---

7. Differences Between TCP and UDP Headers

Header Field	TCP Header	UDP Header
Size	20-60 bytes (variable length due to options)	8 bytes (fixed length)
Fields	Source port, destination port, sequence number, acknowledgment number, data offset, flags, window size, checksum, urgent pointer, options	Source port, destination port, length, checksum
Control Flags	Multiple flags (SYN, ACK, FIN, RST, PSH, URG)	None
Sequence Number	Present	Absent
Acknowledgment Num	Present	Absent

---

8. Difference Between Connection-Oriented and Connectionless Protocols

Aspect	Connection-Oriented (TCP)	Connectionless (UDP)
Connection Setup	Requires handshake (3-way)	No connection setup
Reliability	Guaranteed delivery with acknowledgments	No guarantees
Data Transmission	Stream of bytes, ordered and error-checked	Individual datagrams, may arrive out of order
Overhead	Higher due to connection management	Lower overhead

---

9. Examples of Applications that Use TCP and UDP

Protocol	Example Applications
TCP	HTTP/HTTPS (web browsing), FTP, SMTP (email), SSH, Telnet, DNS zone transfers
UDP	DNS queries, DHCP, VoIP (e.g., Skype), Video streaming, Online gaming, TFTP

If you want me to add commands, packet structure visuals, or dive into practical examples with tools like Wireshark or Nmap for this topic, just ask!